

Seminar am GMD/IPSI Darmstadt :

# Virtuelle Realitäten und andere künstliche Welten

Ausarbeitung des Vortrages

## Amaze

-

# Design of a 3D User Interface to a Database

Version 1.2d SS'96

von Patrick Seiler

Diese Ausarbeitung basiert zum grössten Teil auf dem Buch *Interfaces To Database Systems - Design of a 3D User Interface to a Database* [1] von Boyle, Fothergill und Gray 1994, und auf dem Artikel *Development of a Visual Query Language* [2] von Boyle, Leishman, Fothergill und Gray

Der Seminarvortrag fand am 18.06.1996 am GMD/IPSI Darmstadt statt und wurde von [Patrick Seiler](#), dem Autor dieser Ausarbeitung, gehalten.

Dieses Dokument und die weiteren Unterlagen sind erhältlich unter <http://patrick-seiler.de/projects/virtreal/>

---

## Inhalt

### 1. Einleitung

[1.1 Vorwort](#)

[1.2 Systemhintergrund](#)

### 2. Die vier Schlüsselemente von Amaze

[2.1 Darstellung und Organisation der Daten](#)

[2.2 Konstruktion von Datenbankabfragen](#)

[2.3 Ergebnisdarstellung](#)

[2.4 grafische Einzelergebnis Darstellung](#)

### 3. Beispiele für Datenbankabfragen

[3.1 Universitätsdatenbank](#)

[3.2 Proteindatenbank](#)

### 4. Weitere Arbeiten

[4.1 Auswertungsstudien](#)

[4.2 Erweiterungsmöglichkeiten](#)

[4.3 Weitere 3D Schnittstellen](#)

### 5. Ergebnisse der Diskussion

[5.1 Probleme durch menschliche Faktoren und die Darstellung](#)

[5.2 Probleme durch komplexere Objektklassen](#)

[5.3 Integration der Datenbankanfrage](#)

[5.4 Kernfragen und Anforderungen](#)

6. [Glossar](#)

7. [Literaturverzeichnis](#)

---

# 1. Einleitung

## 1.1 Vorwort

Seit geraumer Zeit ist eine Veränderung im Design von grafischen Benutzerschnittstellen für Datenbanken feststellbar. Herkömmliche Menüs und Dialogboxen werden dabei immer öfter durch Multimedia und s.g. virtuelle Darstellung ersetzt. Nirgendwo ist eine solche Veränderung so notwendig wie im Bereich der Datenbanken.

Die Autoren des Buches "Interfaces To Database Systems" [1] Boyle, Fothergill und Gray, haben erstmals dreidimensionale grafische Techniken für die Oberfläche ihrer objektorientierten Datenbank verwendet.

Mit folgenden Features :

- Verwendung von **animierten interaktiven 3D Grafiken** zusammen mit Standard-Menüs
- Entwicklung einer **3D Darstellung für die Konstruktion der Datenbankanfrage**
- Entwicklung einer **3D Darstellung für die Ergebnis-Repräsentation**

Die steigende Zahl der gelegentlichen Benutzer von Datenbanken, zeigt eine Veränderung vom rein Funktionellen, hin zum Benutzbaren in der Softwareentwicklung. Durch diese neue Generation von Oberflächen, werden Datenbankzugriffe der breiten Masse von Benutzern zugänglich gemacht.

In diesem Papier geht es speziell um die Entwicklung, die Verfahrensweise und das Design von AMAZE. Amaze benutzt high-performance 3D Grafik, als Abfragemöglichkeit für die objektorientierte Datenbank P/FDM.

Der Sinn einer solchen 3D Oberfläche liegt darin, eine bessere Interaktion zwischen dem Benutzer und dem komplexen Computersystem zu bekommen. Während die übliche Desktopdarstellung zu einer einfachen Interaktion mit Textverarbeitungen und Tabellenkalkulationen führt, erleichtert sie nicht gerade die Anfragen an ein Datenbankmanagementsystem. Besonders die Visualisierung der Ergebnisse ist dabei ein Problem.

Herkömmliche Schnittstellen für Datenbanksysteme fallen in zwei Kategorien:

- **Formular basierend**  
Formular basierende Systeme nutzen Tabellen um Ergebnisse darzustellen.
- **Graphen basierend**  
Graphen basierende Systeme hingegen, nutzen Piktogramme / Bilder als Schema für Datenbankabfragen

Amaze erlaubt durch die direkte Darstellung des Datenbankschemas, mit Hilfe von 3D Grafik, dem Nutzer in der Datenbankanfrage hin und her zu navigieren. Eine Anfrage ähnelt so einer Reise von einem Objekt zum anderen. In anderen 3D Ansätzen für Datenbanken ist heute noch ein grosser Unterschied zwischen der Anfragestärke (den Möglichkeiten zur Anfrage) und der Benutzbarkeit der Datenbank festzustellen.

Seit kurzer Zeit existieren noch weitere Arten von Datenbankoberflächen, die auf einer guten visuellen Repräsentation beruhen:

basierend auf 2D Grafik	basierend auf 3D Grafik
InfoCrystal	<a href="#">Lyberworld</a>
Sketch	<a href="#">Amaze</a>

## 1.2 Systemhintergrund

Amaze ist eine grafische Oberfläche für das Datenbankmanagementsystem P/FDM.[4]

P/FDM ist eine Implementation des funktionalen Datenmodells mit objektorientierten Erweiterungen und einer SQL-ähnlichen Abfragesprache [Daplex](#). Dieses Datenbankmanagementsystem basiert auf dem Client-Server Modell. Die Clients sind hier SGI Indigos incl. X11Window/OSFModif, der Server ist eine SUN 4 Workstation.

Die Hauptanwendung ist eine Proteinstruktur-Datenbank für Biochemiker. Diese dient dazu, die komplexen Strukturen von grossen Makromolekülen / [Proteinen](#) verständlich zu machen.

Die Benutzer dieser Datenbank, Biochemiker, sind geübt im Umgang mit 3D Grafikprogrammen, da die Molekular-Darstellungsapplikationen weit verbreitet sind. Der Computer ist ein wichtiges Laborwerkzeug. Daher sind Biochemiker die geeignete Anwendergruppe für die Arbeit mit einer dreidimensionalen Benutzeroberfläche wie Amaze.

## 2. Die vier Schlüsselemente von Amaze

Um das Konzept hinter der Oberfläche besser zu verstehen, folgt nun eine kleine Beschreibung der Funktionalität.

Während der Entwicklung haben sich 4 Schlüsselemente herausgebildet, die eine mögliche Nutzung in der Entwicklung von erweiterten Datenbankabfragesystemen erlauben.

- **Konstruktion von Datenbankanfragen**  
darunter versteht man die Unterstützung des Anwenders bei der Konstruktion einer Anfrage an eine Datenbank. Diese wird hier, im Gegensatz zu SQL, nicht über eine Programmiersprache textuell vom Benutzer eingegeben, sondern mittels dreidimensionaler Grafik interaktiv erstellt.
- **Ergebnisdarstellung**  
Hier wird versucht alle gefundenen Ergebnisse bzw. Lösungen der Anfrage grafisch miteinander in Beziehung zu bringen. Dadurch kann der Anfragesteller eine für sich interessante Lösung auswählen.
- **Grafische Darstellung der Einzelergebnisse**  
Zentrale Frage hier: wie kann der Benutzer der Datenbank ein spezielles Ergebnis untersuchen und bearbeiten, welches er bei der Ergebnisdarstellung ausgewählt hat

An dieser Stelle sind sich die Autoren von Amaze nicht ganz einig. In dem, dieser Ausarbeitung zugrundeliegendem Buch [1], werden nur drei Schlüsselemente erwähnt, in anderen Papieren wie z.B. [2], sprechen die Autoren Boyle, Leishman, Fothergill und Gray aber von vier Elementen. Dies kann möglicherweise am unterschiedlichen Erscheinungszeitpunkt der beiden Papiere liegen. Hinzu kommt in [2] noch die

- **Darstellung und Organisation der Daten**  
als zusätzlicher Punkt

## 2.1 Darstellung und Organisation der Daten

Alle Anfragen an eine Datenbank basieren bei Amaze auf der Navigations-Idee, d.h. der Benutzer navigiert von einem Punkt zum anderen, je nachdem wie sich die Anfrage entwickelt. Farbliche Markierungen erleichtern die Navigation durch die Datenbankanfrage:

### 3 Farben Modell, je eine Farbe für

- die **aktuelle Lösung von Interesse**
- **alle Lösungen mit direkter Beziehung zur aktuellen Lösung**
- **alle anderen Lösungen**

Da aber im allgemeinen auf den Folien und dem Ausdruck dieser Ausarbeitung keinerlei Farben erkennbar sind, gehe ich hier nicht näher auf das Farbenkonzept ein. Ein weiteres Beispiel, in dem die Originalfarben bei der Konstruktion einer Datenbankanfrage erkennbar sind, habe ich zusätzlich im WWW bereitgestellt.

(Anmerkung: dieses Beispiel wurde aus Platzproblemen vom Server gelöscht, ist aber auf Anfrage erhältlich)

Durch die Verwendung der dreidimensionalen Grafik in Amaze, versucht man das 'look and feel' der vorhandenen Molekulargrafikpakete der Biochemiker nachzuahmen. Der Sinn hierbei ist, die vorhandene Erfahrung des Anwenders zu übernehmen und ihn nicht noch zusätzlich zu verwirren.

## 2.2 Konstruktion von Datenbankanfragen

Der Datenbankanfrageteil ist so konstruiert, dass man beliebig viele Anfragen an ein Objekt bzw. Klasse stellen kann. Jede Anfrage wird durch einen Baum unterhalb des entsprechenden Objektes dargestellt. Solch ein Baum besteht aus den üblichen Komponenten, die einen Baum in der Informatik kennzeichnen:

- **Wurzel**  
die Wurzel eines Anfragebaumes ist durch einen Würfel direkt unterhalb der darauf bezogenen Objektklasse dargestellt
- **Blätter**  
Blätter repräsentieren die verschiedenen booleschen Ausdrücke, die der Benutzer der Datenbank für die Anfrage benutzt. Die Eingabe dieser Ausdrücke erfolgt mit Hilfe einer Dialogbox
- **Knoten**  
Die Knoten stehen für logische Operationen zwischen den Boxen

Die Anfrage wird dann mit Hilfe von [Daplex](#) übersetzt und zu einem Remote Prozess weitergeleitet, der für die Verständigung mit der Datenbank zuständig ist. Anfragen können jederzeit neu gestellt oder wiederholt werden.

## 2.3 Ergebnisdarstellung

Ein weiterer wichtiger Punkt neben der Erstellung einer Anfrage, ist die Ergebnisdarstellung.

Für die Darstellung des Ergebnisses wurde die Labyrinthdarstellung gewählt. Diese erlaubt dem Benutzer die Größe des zurückgegebenen Datensatzes zu erkennen und auch welche Ergebnisinstanzen untereinander in Beziehung stehen. Leider ist nicht sichtbar in welcher speziellen Beziehung zueinander.

In der Labyrinthdarstellung wird jede Ergebnisinstanz durch einen Würfel dargestellt und Beziehungen untereinander werden durch Linien gekennzeichnet. Zwischen diesen Ergebniswürfeln kann dann hin und her navigiert werden. Es ist das gleiche Prinzip, wie bei der Konstruktion der Anfragen.

Die Ergebnisse können gruppiert werden, d.h. Ergebnisinstanzen der gleichen Objektklasse können zu s.g. Sets zusammengestellt werden, inklusive einer Farbkodierung. Dabei werden komplette Instanzen durch einen 3D Würfel zusammengruppiert. Vorteil: der Benutzer kann diese Instanzen-Sets einer Objektklasse relativ einfach erkennen.

Jede Instanz dieser Sets ist verbunden mit anderen, direkt in Beziehung stehenden Klassen.

Ein Beispiel für ein Ergebnislabirynth ist auf dem Bild zu sehen:

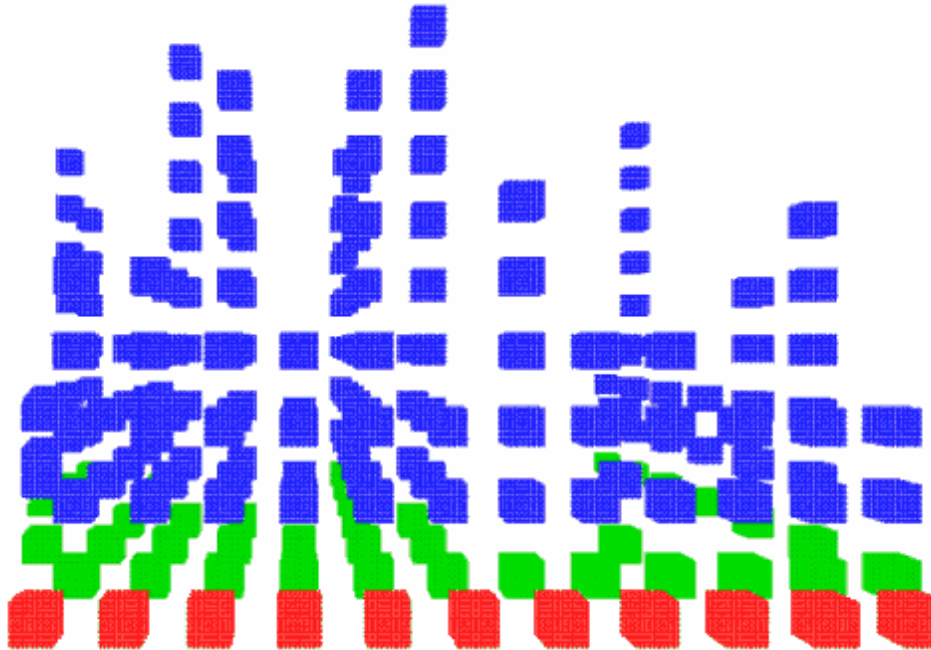


Bild 1: 3D Darstellung des Ergebnislabirynths

Jeder Wert, jede Instanz bei der Eingabekonstruktion, ergibt eine Dimension in der Ausgabe. Dabei können bei komplexen Anfragen natürlich auch mehr als die sonst üblichen 3 Dimensionen entstehen. Aus diesem Grund wird nur ein Teil des Ergebnisraumes wirklich dargestellt.

Genauer, die Daten werden als 3D Schnappschuss des n-dimensionalen Ergebnisses gezeigt. D.h. bei der Navigation von einer Instanz, zu einem in Beziehung stehenden Set von Instanzen, wird die vorhergehende Ergebnisinstanz ausgeblendet und die nächste Dimension an deren Stelle eingeblendet. Dies ist zu verstehen, ähnlich wie wenn jemand um die Ecke geht, bzw. die Richtung in einem Labyrinth ändert.

Eine Beispielanfrage an die Datenbank für ein Ergebnislabirynth könnte lauten:

***suche alle Instanzen der Klasse "Protein", welche ein Molekulargewicht von grösser 10 KDa besitzen - bei allen gefundenen Instanzen zeige die "Sheets" und alle zugehörigen "Strands" an***

Im Ergebnislabirynth dieser Anfrage zeigt eine Dimension die Anzahl der Proteine, eine weitere zeigt die Zahl der Proteinkomponenten für jede einzelne Instanz der Proteine. Jeder einzelne Würfel (Ergebnis) kann dabei als Basis für weitere Anfragen gelten.

#### **Vorteile der 3D Darstellung:**

- Erkenntnis über die Grösse des zurückgegebenen Datensatzes
- Erkenntnis über die Beziehungen untereinander

## Nachteile

- Benutzer mit wenig Erfahrung beim Umgang mit 3D Programmen, haben evtl. Verständnisschwierigkeiten
- Ein einfacher Auszug wie Bild 1, ist relativ schlecht erkennbar

## 2.4 grafische Einzelergebnis Darstellung

Wünscht der Benutzer mehr Informationen über das Ergebnis, welches hinter (bzw. in) einem Würfel verborgen ist, wählt er einfach eine der verfügbaren Darstellungsmethoden aus der Datenbank. Vier spezielle Darstellungsmethoden existieren für die Proteinstruktur-DB:

- **einfache Textdarstellung aller skalaren Werte**
- **3D Protein Molekulardarstellung**  
Darstellung der Struktur des Moleküls (siehe Bild 3), diese übergibt die dreidimensionalen Koordinaten des Atoms und unterstützt Animationen. Die Proteindarstellung ist verbunden mit dem Ergebnislabrynth, d.h. die Bewegungen im Labrynth werden auf die Molekulardarstellung übertragen.
- **phi/psi-Winkel Plotterdarstellung**  
zeigt eine Darstellung der diedral Winkel (zum Erscheinungszeitpunkt des Buches [1], 1994, noch nicht komplett implementiert)
- **Sequenz-Anzeige**  
zeigt die Aminosäuren Rückstände (siehe Bild 2)

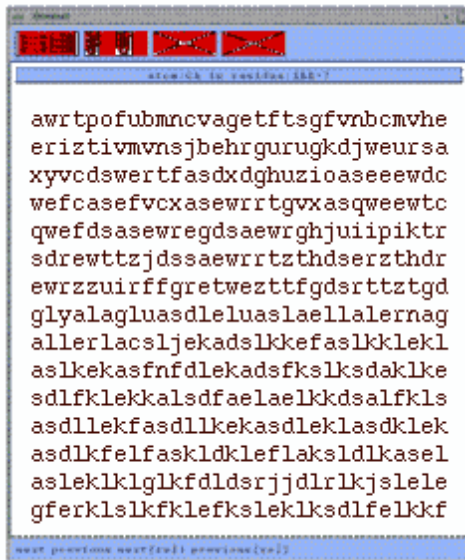


Bild 2: Sequenzdarstellung

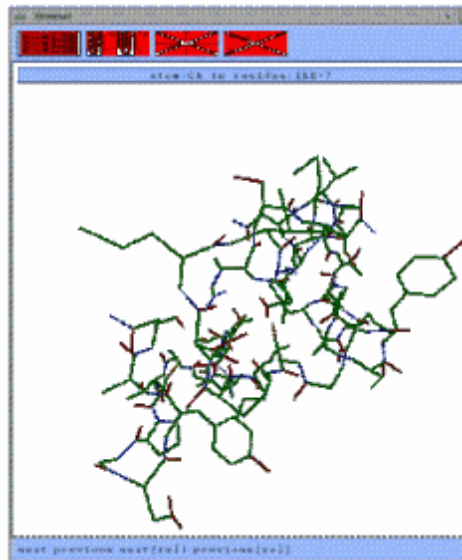


Bild 3: 3D Molekulardarstellung

Der Benutzer kann bestimmte Teile der Darstellung anwählen und bekommt Informationen in Form einer Dialogbox über das Atom und der zugehörigen Aminosäure.

Defaultdarstellung ist die "Textdarstellung aller skalaren Werte", die, wie alle verfügbaren Darstellungsmethoden, in der Datenbank steht. Module mit neuen Funktionen können leicht zur Datenbank hinzugefügt werden, z.B. neue Displaymodule. Die Architektur von Amaze ist extra dafür ausgelegt.

## 3. Beispiele für Datenbankabfragen

Um das Konzept der Konstruktion von Datenbankabfragen besser zu verstehen, folgen nun zwei Beispiele von Datenbankabfragen, die das Konzept von Amaze benutzen: zum einen die schon angesprochene Proteinstrukturdatenbank und zum anderen eine Universitäts Informatikerdatenbank.

### 3.1 Universitätsdatenbank

Eine Beispielanfrage an die Universitätsdatenbank könnte lauten:

**Suche alle Doktoranden, die für Prof. Gray arbeiten und am Datenbank-Visualisierungs-Projekt beteiligt sind.**

Das genauere Vorgehen innerhalb von Amaze sieht dann so aus:

1. zuerst interessiert uns das Objekt **Doktorant** (dargestellt durch eine dreidimensionale Box, wie alle anderen Objekte)
2. zu jeder Box existiert ein objektspezifisches Menü (Bild 4a), hier:
  - **alle Doktoranden** oder
  - eine **speziellere Anfrage** z.B. um die Anzahl genauer zu spezifizieren

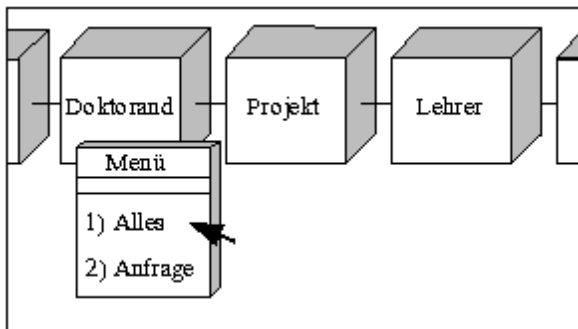


Bild 4a: Objektspezifische Menüs erleichtern die Auswahl aus den verschiedenen zur Verfügung stehenden Objektattributen

3. wähle aus dem Menü **Alle Doktoranden** aus. Dies wird durch eine Box, unter dem aktuellen Objekt hängend, verdeutlicht (Bild 4b). Alle Anfragen werden auf diese Art dargestellt.
4. als nächstes werden alle Doktoranden gesucht, die am Datenbankvisualisierungs Projekt arbeiten, d.h. **Projekt** anwählen.
5. für diese Beispielanfrage interessieren uns nicht **alle** Projekte, deshalb wird eine **speziellere Anfrage** im Menü angewählt.

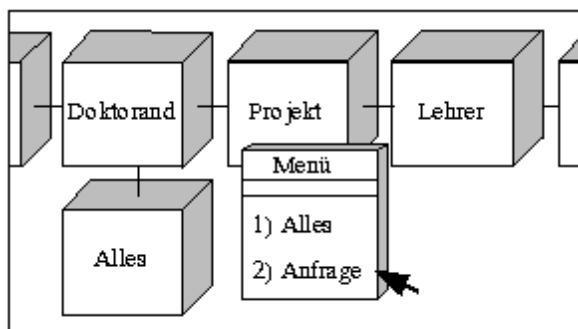
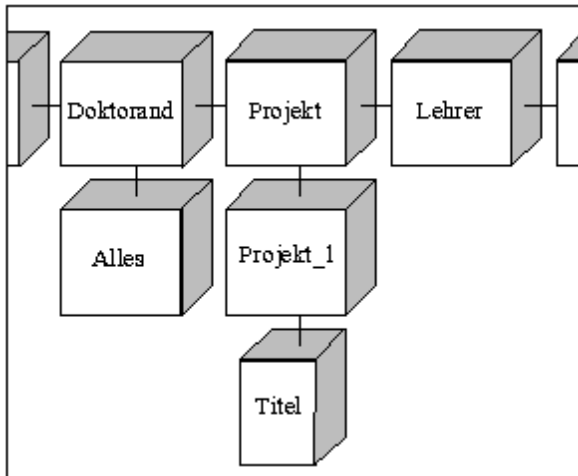


Bild 4b: ein Zwischenschritt auf dem Weg zur fertigen Anfrage

6. in der nun folgenden Dialogbox kann der Benutzer zwischen allen, dem Objekt zugehörigen Attributen, auswählen. Hier wird **Titel** angewählt und

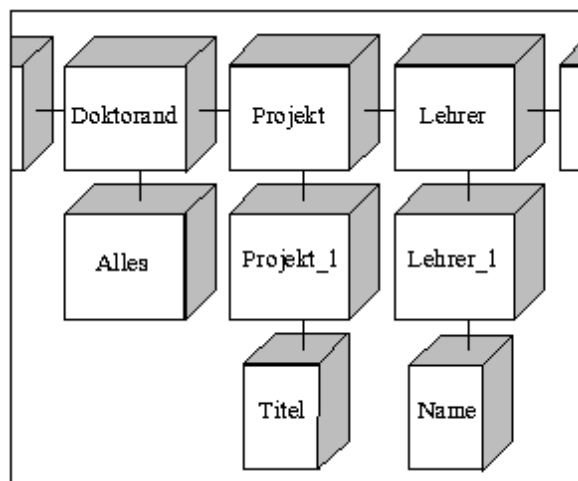
**Datenbankvisualisierung** in die Textbox eingegeben.

7. die Anfrage selbst kann noch einen Namen bekommen, hier **Projekt**. Sinn dieser Namensgebung ist eine eventuelle spätere Speicherung und weitere Verwendung der Anfrage.
8. die aktuell zusammengestellte Anfrage mit der Bezeichnung **project\_1** wird wieder als dreidimensionale Box an die zugehörige Objektklasse **Projekt** angehängt (Bild 4c). Das gewählte Attribut hängt darunter: **Title=Database visualisation** (hier nur als **Titel** dargestellt).



*Bild 4c: Die ausgewählten Attribute werden an die entsprechenden Objekte angehängt*

9. Am Schluss wird genauso mit dem Objekt **Lehrer/Professor** und dem Attribut **name=Gray** verfahren (Bild 4d).



*Bild 4d: Die fertig zusammengestellte Beispielanfrage*

Das Gesamtergebnis unserer Beispielanfrage, mit allen gefundenen Personen, kann wieder wie im [Bild 1](#) mit allen Abhängigkeiten untereinander mit Hilfe von Würfeln dargestellt werden. Beim Anklicken eines der Würfelobjekte, welche in diesem Fall ja die gefundenen Personen darstellen, bekommt man Informationen geliefert, die mit Hilfe eines Toolbrowsers genauer untersucht werden können. Zur Wahl stehen in der Universitätsdatenbank die

- **Textdarstellung der skalaren Werte**, z.B. das Alter oder der Wohnort der Person (siehe dazu auch die Proteindatenbank)
- **ein Bild der gefundenen Person**



Für weitere Datenbanken stehen zusätzliche, neue Darstellungsmethoden zur Verfügung.

## 3.2 Proteindatenbank

Nun noch ein Beispiel zur Verdeutlichung des Konzepts zur Konstruktion von Datenbankabfragen mit Hilfe von Amaze. Diesmal die schon angesprochene Proteinstruktur-Datenbank.

In der Anfrage können natürlich auch mehrere Attribute des selben Objektes mit logischen Operationen verknüpft werden. Als Beispiel dient die Klasse **Biologische Einheit** :

**Suche alle Proteine, welche aus mehr als 200 und weniger als 400 Aminosäuren bestehen**

So wird diese Beispielanfrage von Amaze behandelt:

- Die logische Operation **UND** wird als Tetraeder dargestellt und die Werte alle nebeneinander darunter gehängt. Dies verdeutlicht den Zusammenhang der beiden eingegebenen Werte als eine UND-Verknüpfung:

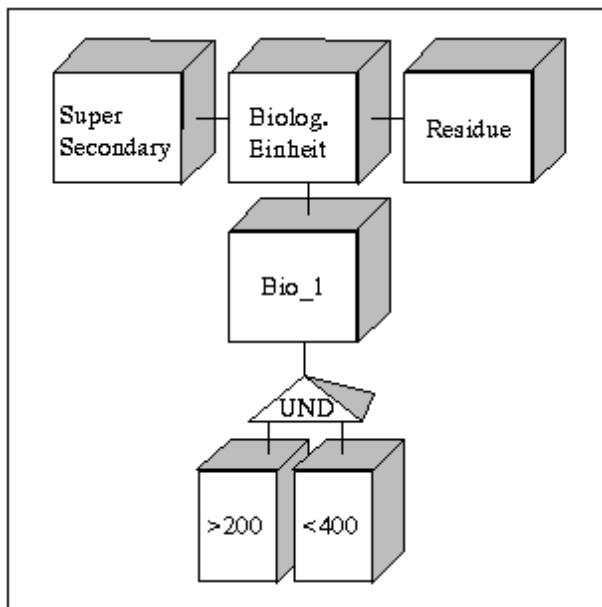


Bild 5: Darstellung logischer Operationen in der Anfrage

- alle gefundenen Proteine werden hintereinander angehängt dargestellt. Das Ergebnis dieser Anfrage würde dann beispielsweise so aussehen, wie in Bild 6a (bei 6 gefundenen Lösungen) :

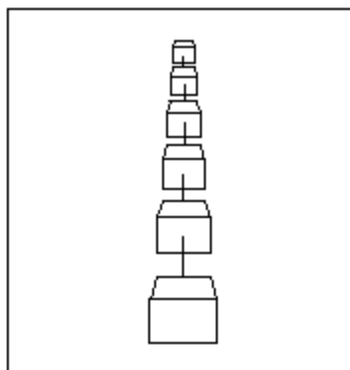


Bild 6a: Alle 6 gefundenen Proteine

- werden noch weitere Werte in die Anfrage eingebracht, so z.B. **"Supersecondary = ALL"**

und "**Residue = ALL**" sieht das Ergebnis dann beispielsweise so aus wie in Bild 6b

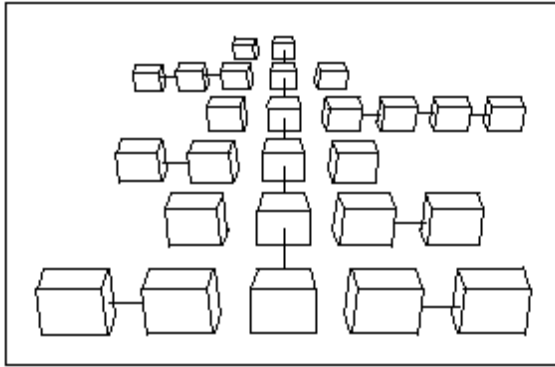


Bild 6b: Die Proteine mit den gefundenen zugehörigen Supersecondary und Residuen

- die Proteine sind natürlich weiterhin in der Mitte dargestellt, die gefundenen **Supersecondary** stehen links von den zugehörigen Proteinen, die **Residuen** rechts der Proteine.  
Die Proteine sind alle durch eine Linie miteinander verbunden, um sie als Objekte des gleichen Typs zu kennzeichnen. Weiterhin sind die Residuen, welche zum gleichen Protein gehören auch untereinander verbunden und direkt neben dem zugehörigen Protein angesiedelt. Gleiches gilt für die Supersecondary.

Nochmal zur Verdeutlichung, die Darstellung der Bilder 6a und 6b dient dazu, **alle** gefundenen Lösungen aufzuzeigen. Sie zeigt aber nicht, welche dieser Lösungen für mich als Betrachter interessanter ist, wie es beim [Bead](#) System der Fall ist. Die Entfernung der Objekte vom Betrachter im 3D-Raum spielt bei Amaze keine Rolle. Alle diese Lösungen sind deshalb gleichwertig.

## 4. Weitere Arbeiten

### 4.1 Auswertungsstudien

Amaze wurde nach 3 Gesichtspunkten entwickelt :

- **Studie über die Benutzbarkeit der Oberfläche**  
Die Brauchbarkeitsstudie ist für das "look and feel" von Amaze zuständig. Hierzu wurden in der Entwicklungsphase der Oberfläche einige Benutzer des Programms hinzugezogen. Der Sinn dieser Studie ist, die bevorzugten Darstellungsmethoden der Anwender herauszufinden.
- **Aufgaben-Studie**  
Die Aufgaben-Studie war zum Erscheinungszeitpunkt von [1] noch in Planung. Hier werden dem Benutzer verschiedene Aufgaben mit variierendem Schwierigkeitsgrad gestellt, die er zu lösen hat. Dabei werden die Geschwindigkeit, die Art, wie der Benutzer die Aufgaben löst und die Probleme die dabei entstehen, aufgezeichnet. Dies dient dazu, die Grenzen der Oberfläche zu erkennen, damit evtl. Schwächen des Interfaces schnell und sicher behoben werden können.
- **Empirische Statistiken**  
Die Empirische-Studie bedeutet eine Auswertung der Statistikergebnisse, die man bei der Interaktion des Anwenders und der Oberfläche erhält. Das bedeutet die Aufnahme des Timings und der Interaktion durch einfache Aufzeichnung aller Benutzereingaben inklusive der Erfolge und Fehler des Benutzers.

Das Feedback aller Studien hilft, den Dialog zwischen dem Benutzer und der dreidimensionalen

Oberfläche zu verbessern. Schliesslich gibt es (noch) nicht so viele 3D Benutzerschnittstellen für Datenbankmanagementsysteme, somit fehlt in diesem Bereich noch etwas die Erfahrung. Die Ergebnisse dienen auch dazu die Entwicklung weiterer 3D Schnittstellen in die richtige Richtung zu lenken.

Frühere Studien zeigten, dass die dreidimensionale Darstellung für den Benutzer sozusagen eine angenehme Art der Unterhaltung darstellt und die Arbeit damit teilweise sogar genossen wird.

## 4.2 Erweiterungsmöglichkeiten

Da die Antwortzeiten von Benutzeroberflächen möglichst gering gehalten werden müssen, haben bei Amaze nur relativ einfache Grafiken Verwendung gefunden. Mit dieser Technik vermeidet man grössere Wartezeiten bei komplexen dreidimensionalen Mappings, dadurch erscheinen die Animationen in Echtzeit. Zusätzliche grafische Features wie

- **Lichteffekte**
- **3D Fonts**
- **Kugeln**

sind zwar vorhanden, müssen aber durch den Anwender bei Amaze speziell angefordert werden.

## 4.3 Weitere 3D Schnittstellen

Neben Amaze gibt es mittlerweile noch weitere 3D Darstellungsmöglichkeiten, die für Datenbanken entwickelt werden. Ein Vertreter dieser Gattung ist Q-PIT.

- **Q-PIT** ist ein experimentelles Anzeigesystem innerhalb der [Dive](#) Umgebung. Mehrere Benutzer können hier innerhalb der gleichen virtuellen Welt umherreisen und sind dabei in der Lage gefundene Daten untereinander auszutauschen. Die Landschaft wird dabei durch die DIVE Umgebung kontrolliert, wobei die einzelnen Attribute der Daten durch Animationen und andere spezielle Techniken auf unterschiedlichste Art und Weise dargestellt werden. Genutzt werden dabei die Position im 3D Raum, die Geschwindigkeit der Rotation, die Grösse, die Farbe und die Form, um die unterschiedlichen Benutzer mit ihren unterschiedlichen Daten erkenntlich zu machen. Q-PIT arbeitet innerhalb einer virtuellen dreidimensionalen Welt, in der der Nutzer, sobald er auf ein Datenobjekt stösst, die Informationen innerhalb dieses Objektes betrachten kann.
- Ein weiterer Vertreter dieser Gattung ist **WINONA**. Winona erlaubt dem Benutzer die Wahl zwischen zwei möglichen Darstellungsarten: der hierarchischen Darstellung und einer s.g. "Circular Wall"-Anzeige. Dabei kann der Benutzer die Daten mit Hilfe eines Controlpanels rotieren lassen und zwischen den zwei Ansichten umschalten. Beide Darstellungsarten sind aber nur für kleinere Datenbanken geeignet.

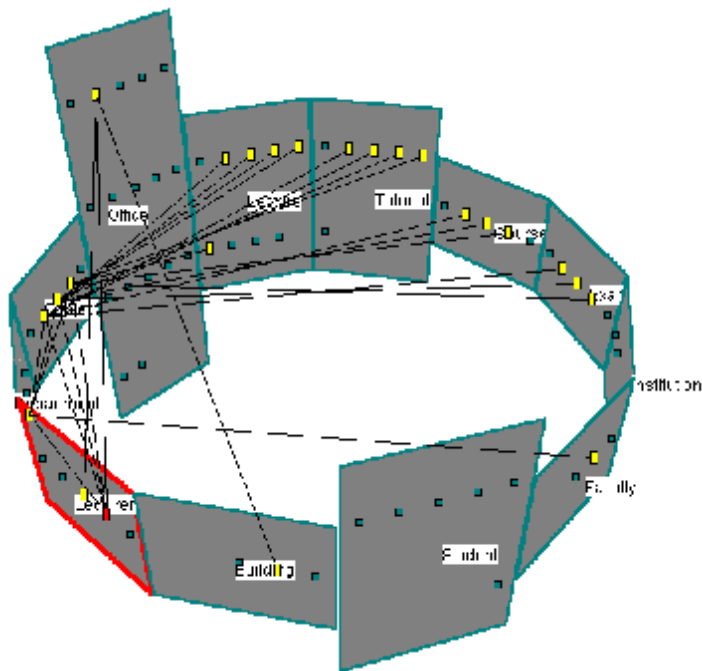


Bild 7: Circular Wall Darstellung in Winona

## 5. Ergebnisse der Diskussion

In diesem Kapitel sind die wichtigsten Punkte, der im Anschluss an den Vortrag erfolgten Diskussion, noch einmal gegliedert dargestellt.

Diese Punkte spiegeln nur z.T. die Meinung der Amaze-Autoren Boyle, Leishman, Fothergill und Gray wieder. In den meisten Fällen ist dies die Meinung des Autors dieser Ausarbeitung. Einige dieser Punkte haben sich erst bei der anschließenden Diskussion mit den Zuhörern des Vortrages ergeben.

### 5.1 Probleme durch menschliche Faktoren und die Darstellung

Probleme entstehen z.T. durch Datenbankmanagementsysteme, menschliche Faktoren, Darstellung und Oberflächendesign.

#### Vorteile gegenüber den üblichen Konzepten :

- **einfacheres Verständnis für Menschen die täglich mit 3D Programmen umgehen**  
für eine herkömmliche Datenbanksprache wie SQL, ist sehr viel mehr theoretisches Verständnis nötig
- **schnelleres Erlernen durch Interaktion**  
spezielle Datenbank-Abfragesprachen zu erlernen dauert für den angestrebten Kundenkreis (Biochemiker...) viel länger als diese Methode, nur wenige verstehen wirklich ausreichend viel von dieser Materie.
- **einfacherer Dialog**  
Vorteil der grafischen Darstellung: man erreicht einen einfacheren Dialog zwischen Anwender und Interface.

#### Nachteile :

- **Zu kleine Arbeitsbereiche**  
zur Zeit besteht noch die Limitierung auf das auf dem Screen sichtbare.

Die Nutzung von virtuellen Umgebungen z.B. Räumen, könnte die Ausdrucksmöglichkeiten erweitern.

- **Konformität ist nicht vorhanden**

im Moment sind nur wenige Ansätze von dreidimensionalen Oberflächen vorhanden. D.h. verschiedene Lösungen für ein ähnliches Ergebnis erschweren das Verständnis. Die Benutzung von mehreren Programmlösungen nebeneinander wird dadurch nicht gerade erleichtert. Ein Standard für 3D Interfaces muss noch geschaffen werden.

- **Lernaufwand**

ohne einen gewissen Lernaufwand geht es auch hier nicht. Dieser Aufwand gestaltet sich aber auf eine ganz andere Art und Weise als beim Lernen einer herkömmlichen Datenbanksprache wie SQL.

Ist das eigentlich ein Nachteil? Für Programmierer, die dieses System benutzen wollen, vielleicht. Diesen ist das Konzept von Amaze teilweise zu unflexibel. SQL wäre dann sicher das geeignetere System. Für Anwender von 3D Programmen sicher nicht.

## 5.2 Probleme durch komplexere Objektklassen

Die Proteinstrukturdatenbank hat weniger als 30 Objektklassen. Welche Probleme ergeben sich mit weitaus komplexeren Klassen?

**Probleme :**

- evtl **unübersichtliche Darstellung**
- **zu viele Dimensionen** bei grösseren Anfragen  
gibt es vielleicht bessere Verfahren für n-dimensionale Darstellungen als 3D-Snapshots?
- **lange Navigationspfade** von einem Objekt zum anderen

## 5.3 Integration der Datenbankanfrage

Was bringt die Integration der Datenbankanfrage in das Ergebnislabirynth?  
Bisher werden die Anfrage und das Ergebnis in verschiedenen Fenstern dargestellt.

**Vorteile der Integration in das Ergebnislabirynth:**

- **einfache dynamische Erweiterbarkeit** der Ergebnisse um weitere Anfragen
- **der Anwender lebt** dann sozusagen **im Ergebnisraum**
- Möglichkeit **weiterer Anfragen vom aktuellen Standpunkt** aus
- **Darstellung der Anfrage zwischen den aktuellen Ergebnissen**

**Nachteil :**

- **noch unübersichtlicher?**  
wird dann evtl. alles noch unübersichtlicher bei komplexen Datenbanken (oder Anfragen) als dies jetzt schon der Fall sein kann?

## 5.4 Kernfragen und Anforderungen

Was sind die Kernfragen die an ein solches System gestellt werden sollten? Welche Anforderungen muss es erfüllen?

- **wie schnell kann man sich in das System einarbeiten ?**  
Dauert es zu lange bis man den Syntax der Anfragen verstanden hat, wird es nur sehr wenige Benutzer finden.
- **wie schnell kommt man mit einer Anfrage zum Ergebnis ?**  
Ähnlich dem Punkt vorher, zu kompliziert darf das Verfahren bei den Anfragen nicht sein. Dies ist natürlich vom gewünschten Benutzerkreis abhängig, aber prinzipiell überall gültig.
- **wie werden die gefundenen Ergebnisse dargeboten ?**  
Dieser Punkt ist m.E. bei der Entwicklung von Amaze ein wenig vernachlässigt worden. Die Darstellung der Ergebnisse erlaubt zwar das Erkennen von Beziehungen untereinander, aber wie diese Beziehung genau aussieht kann man nicht erkennen. Daran sollte man noch etwas arbeiten.
- **wieviele Möglichkeiten zur Einzelergebnisdarstellung gibt es ?**  
Eine ausreichende Anzahl guter Möglichkeiten zur Darstellung von Einzelergebnissen ist Voraussetzung, damit eine solche Datenbank überhaupt Verwendung findet

## 6. Glossar

### Proteine

Proteine sind lineare Moleküle, zusammengesetzt aus sich wiederholenden Aminosäuren. Diese bilden eine Protein-Kette, deren dreidimensionale Form ist eine Darstellung der Eigenschaften des Proteins.

Viel mehr muss ein Informatiker davon nicht verstehen, um hinter das Konzept von Amaze zu blicken. Deshalb sollte man sich durch die Verwendung von Fachausdrücken aus dem Biochemiebereich nicht stören lassen.

### P/FDM

P/FDM = Prolog / Functional Data Model

P/FDM ist ein Datenbankmanagementsystem des funktionalen Datenmodells mit objektorientierten Erweiterungen und einer SQL-ähnlichen Abfragesprache.

Weitere Informationen zu P/FDM unter: <http://www.csd.abdn.ac.uk/~pfdm>

### Daplex

Daplex ist eine High-Level Datenbankabfrage und manipulations -Sprache basierend auf Prolog. Sie erlaubt auch Nicht-Experten schwierige Ausdrücke zusammenzustellen, ohne erklären zu müssen, wie das gewünschte Ergebnis berechnet werden soll. Daplex ersetzt hier zwar SQL, unterscheidet sich aber doch in einigen Punkten davon, z.B ist ein Optimiser für die Optimierung von Anfragen zuständig und übersetzt sie später nach Prolog.

Weitere Informationen zu Daplex in [2]

### Dive

Steht für Distributed Interactive Virtual Environment.

Dive ist ein auf dem Internet basierendes Multiuser VR System, bei dem alle Teilnehmer gleichzeitig in einem 3D Raum navigieren. Die verschiedenen Benutzer können sich dabei sehen, treffen und zusammenarbeiten.

Weitere Informationen unter: <http://sics.se/dce/dive/dive.html>

### Bead

Informationen zu Bead sind im Zuge des Seminars "Virtuelle Realitäten und andere

künstliche Welten" durch einen eigenen Vortrag ausreichend vorhanden. Weitere Informationen unter: <http://bolli.home.pages.de>

---

## 7. Literaturverzeichnis

- [1]:  
Interfaces To Database Systems -  
Design of a 3D user Interface to a Database  
John Boyle, J. Fothergill und Peter M.D. Gray, 1994
  - [2]:  
Development of a Visual Query Language  
<http://www.biochem.abdn.ac.uk/~john/vlq/vlq.html>  
John Boyle, Scott Leishman ,J. Fothergill und Peter M.D. Gray
  - [3]:  
Graphical Interfaces to Database systems  
<http://www.biochem.abdn.ac.uk/~john/if2db.html>
  - [4]:  
Aberdeen University P/FDM Homepage  
<http://www.csd.abdn.ac.uk/~pfdm>
  - [5]:  
DIVE - The Distributed Interactive Virtual Environment  
<http://sics.se/dce/dive/dive.html>
  - [6]:  
Winona - a PC based 3D graphical interface a Database  
<http://www.biochem.abdn.ac.uk/~john/winona.html>
  - [7]:  
John Boyle  
<http://www.biochem.abdn.ac.uk/~john/>
  - [8]:  
Scott Leishman  
<http://www.biochem.abdn.ac.uk/~scott/scott.html>
  - [9]:  
John Fothergill  
<http://www.biochem.abdn.ac.uk/~jef/jef.html>
  - [10]:  
Peter M. D. Gray  
<http://www.csd.abdn.ac.uk/~gray/home-page.html>
- 

Copyright: Patrick Seiler, 1996